Báo cáo về bài thực hành: Thực hành ứng dụng ECC vào OpenSSL để bảo vệ tính toàn vẹn

Mục lục

[1.1. Nội dung và hướng dẫn bài thực hành 1](#_Toc184106044)

[1.1.1. Mục đích 1](#_Toc184106045)

[1.1.2. Yêu cầu đối với sinh viên 1](#_Toc184106046)

[1.1.3. Nội dung thực hành 1](#_Toc184106047)

[1.1.4. Phân tích yêu cầu 3](#_Toc184106048)

[1.1.5. Thiết kế bài thực hành 3](#_Toc184106049)

[1.1.6. Cài đặt và cấu hình các máy ảo 4](#_Toc184106050)

[1.1.7. Tích hợp và triển khai 6](#_Toc184106051)

[1.1.8. Thử nghiệm và đánh giá 7](#_Toc184106052)

1. Bài thực hành: Ứng dụng ECC vào OpenSSL
   1. Nội dung và hướng dẫn bài thực hành
      1. Mục đích

Giúp sinh viên tìm hiểu và ứng dụng ECC(Elliptic Curve Cryptography), sử dụng OpenSSL để trao đổi khoá, tạo và xác minh chữ ký số.

* + 1. Yêu cầu đối với sinh viên

Có kiến thức về mật mã và các công cụ liên quan

* + 1. Nội dung thực hành

Khởi động bài lab, vào terminal gõ:

labtainer –r ecc-integrity

*(chú ý: sinh viên sử dụng mã sinh viên của mình để nhập thông tin email người thực hiện bài lab khi có yêu cầu, để sử dụng khi chấm điểm)*

Sau khi khởi động xong hai terminal ảo sẽ xuất hiện lần lượt là Alice và Bob với password là ubuntu .

Trên máy Alice sử dụng OpenSSL để tạo cặp khoá công khai và khoá bí mật.

*openssl ecparam -name prime256v1 -genkey -noout -out alice\_private\_key.pem*

*openssl ec -in alice\_private\_key.pem -pubout -out alice\_public\_key.pem*

Trên máy Bob tương tự để tạo cặp khoá công khai và khoá bí mật

*openssl ecparam -name prime256v1 -genkey -noout -out bob\_private\_key.pem*

*openssl ec -in bob\_private\_key.pem -pubout -out bob\_public\_key.pem*

Sau đó sử dụng SFTP để vận chuyển khoá công khai giữa 2 máy cho nhau

Alice tính khóa bí mật dùng chung với khóa công khai của Bob:

*openssl pkeyutl -derive -inkey alice\_private\_key.pem -peerkey bob\_public\_key.pem -out alice\_shared\_secret.bin*

Bob tính khóa bí mật dùng chung với khóa công khai của Alice

*openssl pkeyutl -derive -inkey bob\_private\_key.pem -peerkey alice\_public\_key.pem -out bob\_shared\_secret.bin*

Nếu thực hiện đúng, khoá *alice\_shared\_secret.bin* và *bob\_shared\_secret.bin* sẽ giống nhau và được sử dụng để mã hoá

Sau khi trao đổi khóa thành công, sinh viên sử dụng ECDSA để ký số dữ liệu và xác minh tính toàn vẹn.

Alice tạo chữ ký số cho file filetoview.txt đã có sẵn

*openssl dgst -sha256 -sign alice\_private\_key.pem -out <signature\_file(bin)> filetoview.txt*

Sau đó chuyển file cùng với file chữ ký số sang cho Bob với SFTP

Sau đó Bob xác minh chữ ký số

openssl dgst -sha256 -verify alice\_public\_key.pem -signature <sigature\_file> <file \_name>

Sinh viên cần xác cả thực thành công lấn thất bại để hoàn thành task

Kết thúc bài lab:

stoplab ecc-integrity

Khởi động lại bài lab

labtainer –r ecc-integrity

Kiểm tra bài lab

checkwork ecc-integrity

* + 1. Phân tích yêu cầu

Bài thực hành gồm 2 máy tính nằm trong cùng mạng LAN. Trong đó 1 máy là alice,máy còn lại là bob. Để hoàn thành bài thực hành sinh viên cần sử dụng openssl để sinh khoá, mã hoá, giải mã, tạo và xác thực chữ ký số khi truyền file giữa 2 máy.

* + 1. Thiết kế bài thực hành

Trên môi trường máy ảo Ubuntu được cung cấp, sử dụng docker tạo ra 2 container: 1 container mang tên “alice” và 1 container mang tên “bob”.

Tạo một mạng LAN có cấu hình: 192.168.20.0/24 và gateway: 192.168.20.1

Cấu hình docker gồm có:

alice: Địa chỉ trong mạng LAN 192.168.20.3 và gateway: 192.168.20.1

bob: Địa chỉ trong mạng LAN 192.168.20.4 và gateway: 192.168.20.1

config: lưu cấu hình hoạt động của hệ thống

dockerfiles: mô tả cấu hình của 2 container: alice và bob, trong đó:

alice: sử dụng các thư viện mặc định hệ thống và openssl giúp thực hiện các lệnh và chức năng của tool này.

bob sử dụng các thư viện mặc định hệ thống và openssl giúp thực hiện các lệnh và chức năng của tool này.

docs: lưu phần mô tả hướng dẫn làm bài thực hành cho sinh viên.

Các nhiệm vụ cần phải thực hiện để thực hành thành công:

Sử dụng openssl để tạo cặp khoá bí mật và công khai ở máy alice và bob, màn hình hiện chữ “writing EC key” vì bài lab đã được cài đặt chấm điểm dựa vào vết này.

Tạo chữ ký số cho file “filetoview.txt” ở máy alice bằng khoá đã tạo sau đó mã hoá file để gửi cho máy bob

Xác thực file đã giải mã với file chữ ký đã gửi từ trước và màn hình hiện ra “OK” vì bài lab đã cài đặt chấm điểm dựa vào vết này.

Xác thực file đã giải mã với file chữ ký đã gửi từ trước và màn hình hiện ra “Failed” vì bài lab đã cài đặt chấm điểm dựa vào vết này.

Kết thúc bài lab và đóng gói kết quả.

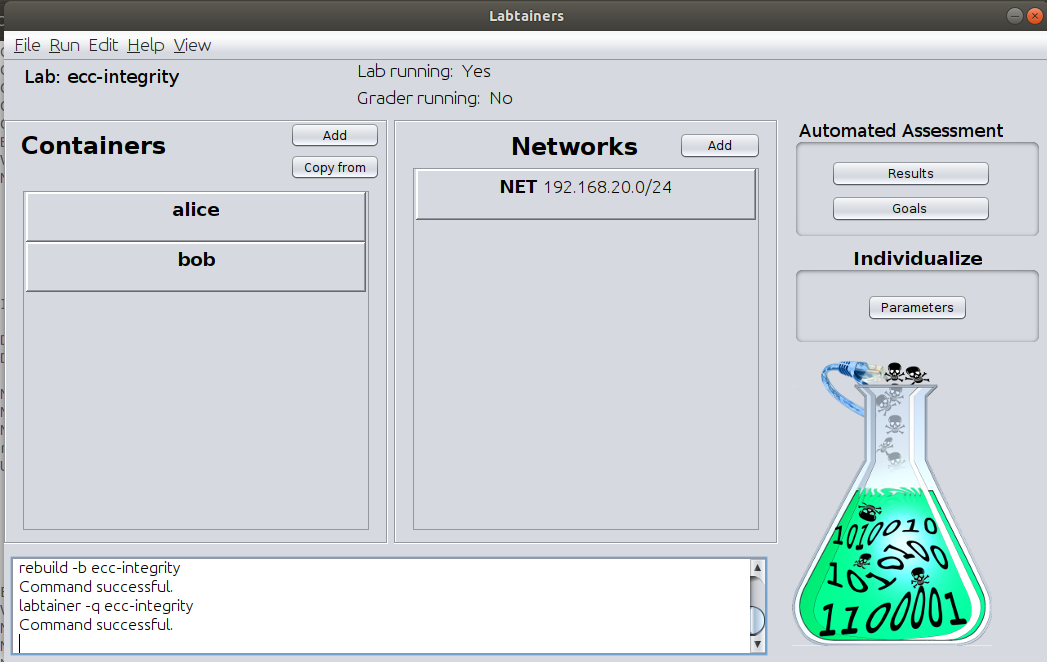
Thiết lập hệ thống mạng sao cho máy attacker và máy victim cùng một mạng LAN.

Để đánh giá được sinh viên đã hoàn thành bài thực hành hay chưa, cần chia bài thực hành thành các nhiệm vụ nhỏ, mỗi nhiệm vụ cần phải chỉ rõ kết quả để có thể dựa vào đó đánh giá, chấm điểm. Do vậy, trong bài thực hành này hệ thống cần ghi nhận các thao tác, sự kiện được mô tả và cấu hình như bảng 1

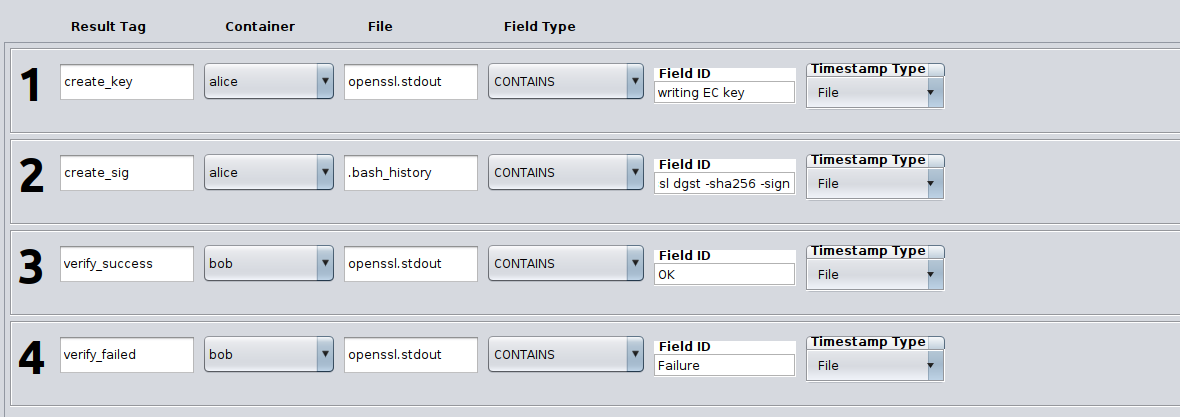
Bảng 1. Bảng Result

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Result Tag | Container | File | Field Type | Field ID | Timestamp Type |
| create\_key | bob | openssl.stdout | CONTAINS | Writing EC key | File |
| create\_sig | alice | .bash\_history | CONTAINS | openssl dgst -sha256 -sign | File |
| verify\_success | bob | openssl.stdout | CONTAINS | OK | File |
| verify\_failes | bob | openssl.stdout | CONTAINS | Failure | File |

* create\_key: hiển thị thông báo tạo cặp khoá ECC thành công
* create\_sig: tạo chữ ký số thành công
* verify\_success: hiển thị thông báo xác thực thành công file
* verify\_failed: hiển thị thông báo xác thực thành công file
  + 1. Cài đặt và cấu hình các máy ảo



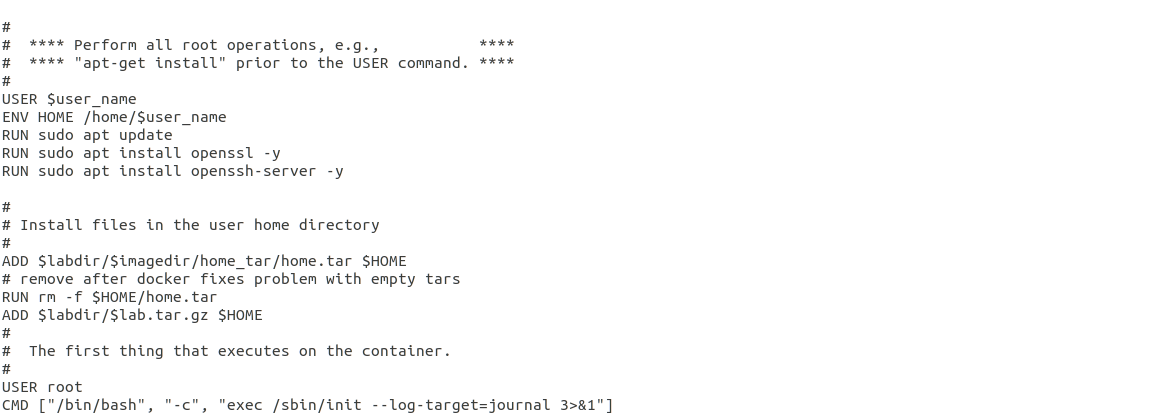
Hình 1. Giao diện Labedit của bài lab



Hình 2. Cài đặt phần Result



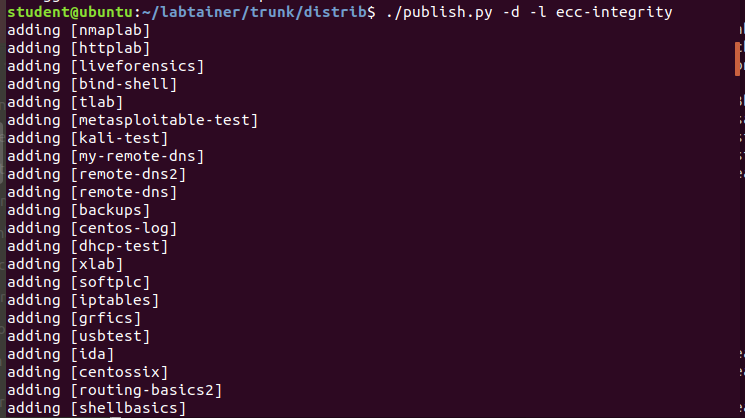
Hình 3. Dockerfiles của máy alice



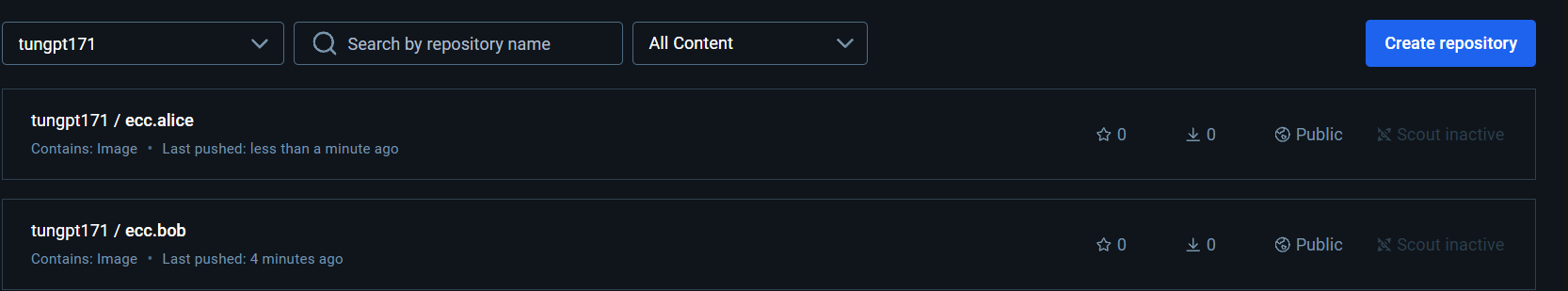
Hình 4.Dockerfiles của máy bob

* + 1. Tích hợp và triển khai
       1. Dockerhub

<https://hub.docker.com/u/tungpt171>



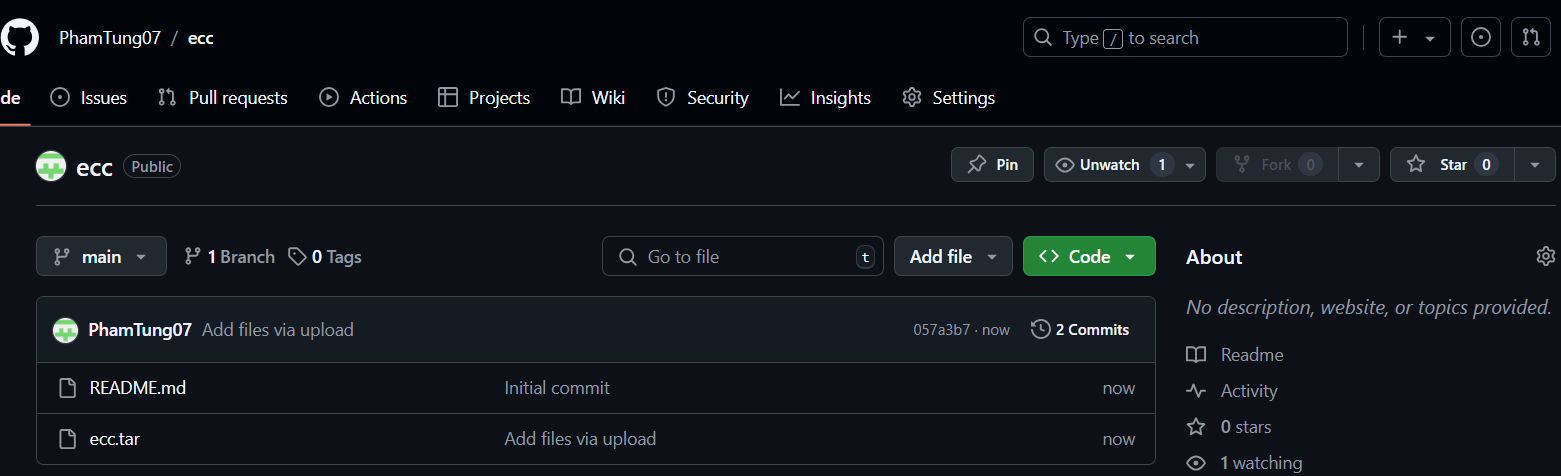
Hình 5.Đảy các container lên Dockerhub



Hình 6.Các container được đẩy lên Dockerhub

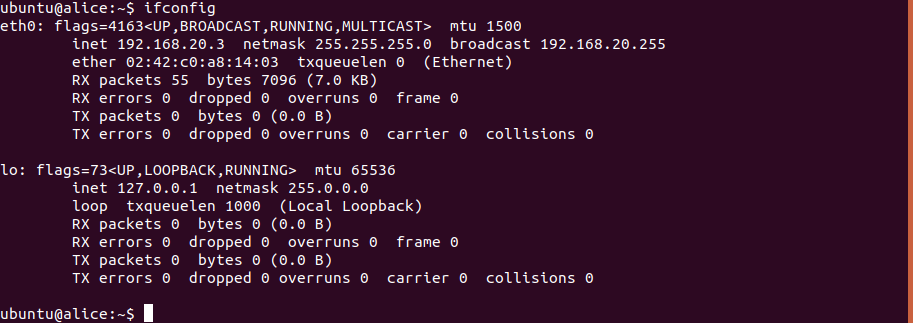
* + - 1. Github

<https://github.com/PhamTung07/ecc>

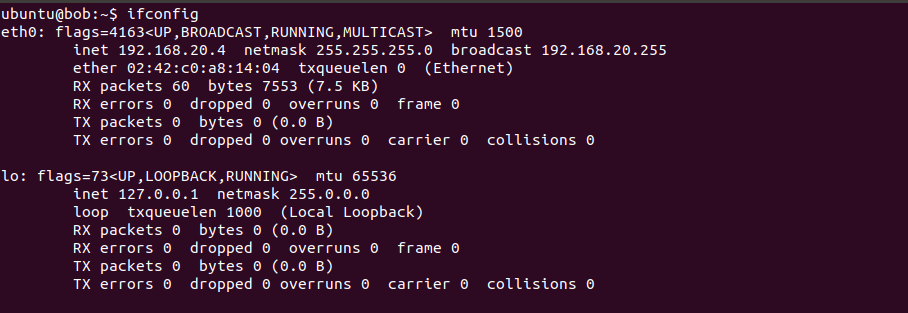


Hình 7.File ecc.tar chứa bài thực hành

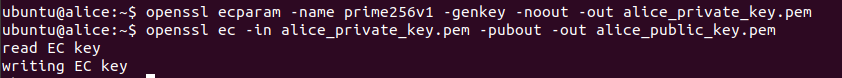
* + 1. Thử nghiệm và đánh giá



Hình 8.IP của máy alice



Hình 9.IP của máy bob



Hình 10.Tạo cặp khoá ECC



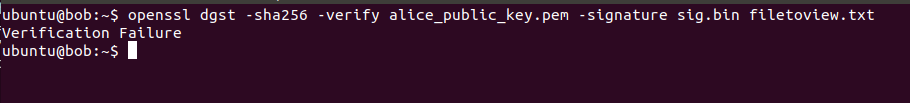
Hình 11.Tạo khoá chung



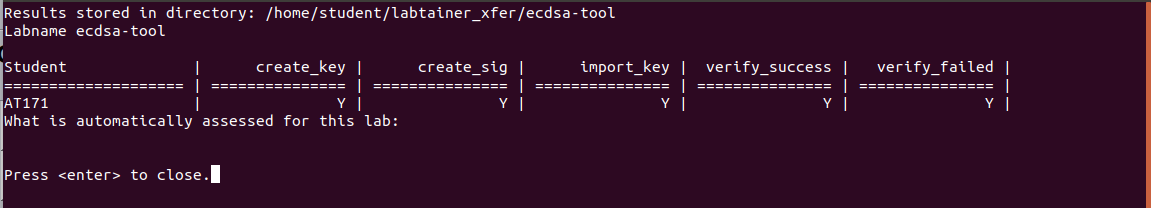
Hình 12.Tạo chữ ký số



Hình 13. Xác thực thành công chữ ký số



Hình 14. Xác thực thất bại chữ ký số



Hình 15. Đánh giá kết quả bài thực hành